Improving Intrusion Detection System Based on Snort Rules for Network Probe Attacks Detection with Association Rules Technique of Data Mining

Nattawat Khamphakdee[1], Nunnapus Benjamas[1] & Saiyan Saiyod[2]

[1]Advanced Smart Computing Laboratory
[2]Hardware-Human Interface and Communications Laboratory
Department of Computer Science, Faculty of Science, Khon Kaen University
123 Moo 16 Mittapap Rd., Nai-Muang, Muang District, Khon Kaen, 40002 Thailand
Email: nunnapus@kku.ac.th

Abstract. The intrusion detection system (IDS) is an important network security tool for securing computer and network systems. It is able to detect and monitor network traffic data. Snort IDS is an open-source network security tool. It can search and match rules with network traffic data in order to detect attacks, and generate an alert. However, the Snort IDS can detect only known attacks. Therefore, we have proposed a procedure for improving Snort IDS rules, based on the association rules data mining technique for detection of network probe attacks. We employed the MIT-DARPA 1999 data set for the experimental evaluation. Since behavior pattern traffic data are both normal and abnormal, the abnormal behavior data is detected by way of the Snort IDS. The experimental results showed that the proposed Snort IDS rules, based on data mining detection of network probe attacks, proved more efficient than the original Snort IDS rules, as well as icmp.rules and icmp-info.rules of Snort IDS. The suitable parameters for the proposed Snort IDS rules are defined as follows: Min_sup set to 10%, and Min_conf set to 100%, and through the application of eight variable attributes. As more suitable parameters are applied, higher accuracy is achieved.

Keywords: *apriori algorithm; data mining; intrusion detection system; network probe attack; network security; Snort IDS rules.*

1 Introduction

Internet technology has become an important part of daily communication through e-mail, social media interaction, e-learning, and so on. Additionally, corporations large and small have expanded upon basic Internet communication to include intercompany correspondence, direct consumer marketing, and internet shopping. Of course, risks are incurred as inefficient and ineffective security tools invite attacks from Internet hackers. Prevention technology, such as firewalls, antivirus programs, and malicious software removal programs, do not provide absolute protection as attackers apply new techniques to assault the network and its users [1]-[3].

Received December 30th, 2014, Revised February 23rd, 2015, Accepted for publication March 4th, 2015.
 ISSN: 2337-5787, DOI: 10.5614/itbj.ict.res.appl.2015.8.3.4

Network attacks may be divided into four categories [4]: denial of service (DoS) attacks, user to root (U2R) attacks, remote to local (R2L) attacks, and network probe attacks. DoS attack is characterized by the attacker interrupting or denying the user access to a server. Examples of DoS attacks include the *Mailbomb*, *Ping of Death*, and *Neptune*. U2R attacks allow an unseen 'privileged access' by extending the permissions of the root, such as those of an administrator. The 'buffer overflow' attack is the most common form of the U2R attack. R2L attacks invade the target system without the owner's permission. Lastly, network probe attacks gather and analyze information in order to map the network system. Scanning software programs, such as *Nmap*, *Mscan*, and *Satan*, collect information from the network target system (IP address, host name, operating system (OS), and service application). Although, network probe attacks only collect data, the basic information obtained may be used in future attacks of other kinds. This paper focuses on network probe attacks.

Currently, intrusion detection systems (IDS) play an important role in maintaining network security. An IDS detects and monitors network traffic data on network systems and alerts the user when a malicious attack occurs [5]. IDS utilizes rules to search and match network traffic data. There are two techniques for intrusion detection: anomaly-based intrusion detection, and misuse-based intrusion detection. Furthermore, IDS can be divided into two types: host-based intrusion detection systems (HIDS) and network-based intrusion detection Systems (NIDS) [2],[6]. Limitations in maintaining network security are based on the IDS's accuracy, self-adaptability, and scalability [7].

Data mining has been applied within IDS [8] in an attempt to improve accuracy and efficiency, as it can search, analyze, and process discovered information in large amounts [9]. Data mining techniques consist of three parts: classification, clustering, and association rules [10].

In this paper, we propose the improvement of the intrusion detection system based on Snort rules for network probe attack detection, utilizing the association rules technique of data mining. The association rule technique analyzes the network traffic data pattern, after which Snort IDS rules are generated according to the network's traffic data behavior.

In Section 2, we briefly discuss the Snort IDS and data mining technique as applied to the IDS. In Section 3, the backgrounds of the Snort IDS, association rules, and the Apriori algorithm are described. The system architecture is presented in Section 4, and Section 5 describes the procedure for experimental evaluation. Lastly, the conclusion and recommendations for future work are presented in Section 6.

2 Related Work

Reviewing several papers that discuss Snort IDS through data mining we find the explanation and implementation of intrusion detection systems utilizing a Snort-based IDS within the Linux operation system [1],[11]. Within this operating system alerts were generated based on the results of Snort IDS through the utilization of the Basic Analysis and Security Engine (BASE). This effort assists the administrator in the analysis of the network's internet connection. The intrusion detection system was further implemented with Snort and WinPcap, within the Windows operating system [12] and a firewall was configured, based on the Windows operating environment. The Snort IDS rules, however, were not improved.

The Snort IDS rules for intrusion detection of network probe attacks were improved through the utilization of the MIT-DARPA 1999 dataset in weeks four and five [5]. The authors analyzed the network traffic data of the attack by applying the Wire Shark software to the dataset. They further compared the detection performance of the network probe attacks with the Detection Scoring Truth. However, the pattern of the network probe attacks in the dataset required additional time to analyze.

Further utilization of Snort IDS involved monitoring web content in real-time for any abnormal behavior patterns within a campus network [13]. The results generated in the alert analysis of the Snort IDS, in high-speed networks within a campus network [14] demonstrated the highest detection of ICMP PING attacks. In designing a Snort IDS model to analyze and pattern match the protocol (in order to improve the speed and accuracy of the intrusion detections system in the campus network), ACID (Analysis Center Intrusion Detection) was chosen to display alarm information [15]. The Snort IDS rules, however, were not improved.

Data mining techniques and data mining algorithms involved in the intrusion detection system and the architecture of the data mining have also been proposed [7]. A model of an intrusion detection system based on data mining is presented and described further in [16]. The authors also proposed modules consisting of a data collection module, preprocessing module, detection module, and response module. Thus, the design and implementation of an intrusion detection system based on data mining utilizing the Apriori algorithm [17] were executed. The experiment resulted in the efficient detection of new types of attacks. Additionally, an intrusion detection system based on data mining technique on database was further introduced [18]. The application of the Apriori algorithm extracted user behavior patterns and generated a greater efficiency of the Snort IDS Rules.

3 The Background

We discuss the background of Snort IDS, the applicable Association Rules Mining, and the process of the Apriori algorithm in three parts.

3.1 The Background of Snort IDS

Matin Roesch developed the Snort Intrusion Detection and Prevention System (Snort IDS/IPS) in 1998, using C language. It has been downloaded as a network security tool over four million times, by approximately 500,000 registered users [19]-[20]. As an open-source, lightweight software application Snort can be installed on numerous operating systems (Linux, Windows, etc.) in almost all forms of computer architecture. It can analyze the network traffic data in real-time by utilizing the different existing rules of the attack. Snort's components are: packet decoder, preprocessors, detection engine, logging and alerting systems, and output modules [12]-[13]. Snort utilizes existing rules, which are patterns of known attacks for searching and matching the network traffic data. If an abnormal behavior pattern is detected it generates an alert. The structure of Snort rules consists of two logical parts [13],[21]. The first part is the rule header, while the second is the rule option, as shown in Figure 1.

The rule header contains the following fields: action, protocol, source address, source port, direction, destination address, and destination port, as shown in Figure 2. Furthermore, the action field in a Snort rule has three properties: alert, log, and pass. The protocol field acts as the criteria where to detect network traffic data, which include IP, TCP, UDP, and ICMP. The Snort rules contain two IP addresses: (1) source and (2) destination. These fields specify the host IP address. If we wish to detect *every* IP address, the field of the IP address is defined as "any". The direction field determines the direction of the network traffic data between source and destination, defined as "->", "<-" and "< >". The port field examines the criteria within the Snort rule to determine the port of protocol. Again, if we wish to determine every port, the field of the port is defined as "any" [22].

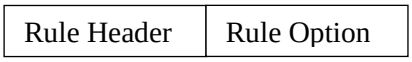

Figure 1 Structure of the Snort IDS Rule.

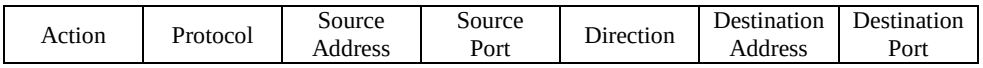

Figure 2 Structure of Snort IDS Rule Header.

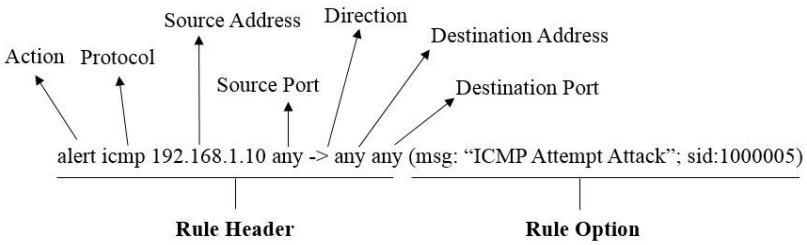

Figure 3 Example of Snort IDS Rule.

The rule options of Snort consist of two parts: a keyword and an argument (defined inside parentheses and separated by a semicolon). The keyword options are separated from the argument by a colon. A Snort rule can have one (or multiple) options within a rule option. Examples of keyword options are msg, ttl, tos, and icode.

An example of the aforementioned Snort IDS rule is shown in Figure 3. If it matches the network traffic data of an ICMP protocol field, such as source address, source port, destination address, and destination port, for example 192.168.1.10, any, any, any, respectively, an alert is generated that outputs the message *ICMP Attempt Attack* with the signature ID 1000005.

3.2 Background of Association Rules Mining

Generally speaking, the various techniques of data mining can be divided into three types: association rules mining, classification, and clustering [10], of which association rules mining is the most important type [23]. First introduced in Agrawal, *et al*., 1993 [24], association rules mining is the process of extracting relevant information from a large database or data warehouse. This may be applied to several research areas, such as medicine, risk management, and business. In order to understand the formal statement of the association rules problem, the definition [25] can be described as follows:

Let $I = \{i_1, i_2, i_3,\ldots, i_n\}$ be a set of items. Let $D = \{T_1, T_2, T_3,\ldots, T_m\}$ be a set of the transection T, where T is a unique transaction number called *TID*; and each transaction T in database D is a set of items I, such that $T \subseteq I$. This $X \Rightarrow Y$ rule is an implication form of the association rules mining, where $X \subset I$ and $Y \subset I$ then $X \cap Y = \varnothing$. X is referred to as the antecedent item of the association rules (left-hand-side or LHS), whereas Y is referred to as a consequent item (right-hand-side or RHS) [3] in which the parameters are of great importance. The minimum support threshold values (Min_sup) and the minimum confidence threshold value (Min_conf), respectively, are expressed as a percentage. The

Min_sup value is used to count each item set in the database. If an item set has a value greater than or equal to the Min_sup value (as specified by the user), it will be a frequent item set. The Min_conf value measures the strength of the association rule of the entire database transaction containing both X and Y. Likewise, if the association rules have a value greater than or equal to the Min_conf value, again user-specified, it will be a frequent rule. The Min_sup and the Min_conf values are calculated using Eqs. (1) and (2) [3],[23], respectively.

$$\text{Support}(\text{X}) = \frac{\text{Support count}(\text{X})}{\text{Total number of Transection in D}} \tag{1}$$

$$\text{Support}(\text{X} \Rightarrow \text{Y}) = \frac{\text{Support count of}(\text{X} \Rightarrow \text{Y})}{\text{Support count of}(\text{X})} \tag{2}$$

3.3 Concept of the Apriori Algorithm

Agrawal and Srikant proposed the Apriori algorithm in 1994 [25]. In order to find out the frequent item set in the large database, two processes are executed. First, the Apriori algorithm generates the candidate item sets (C_k) and calculates the support count of each item. Afterwards it generates the frequent item set (L_k) and prunes the item sets which fall below the Min_sup value.

The Apriori algorithm also determines the frequent k-item set (L_k). First, it calculates the support count of each item set to generate the candidate item sets (C_1). Then, it prunes C_1 of elements lower than the Min_sup value, allowing the frequent item set L_1 to be generated. Then, it determines the frequent 2-item sets (L_2), which are generated from joining L_1 with the candidate 2-itemsets (C_2). Again, pruning C_2 of elements lower than the Min_sup value the frequent item sets (L_2) is generated. This process is repeated until frequent k-item sets can no longer be generated. However, the Apriori algorithm properties dictate that every subset (k-1) within the frequent k-item set must have originated from the frequent item sets.

4 System Architecture

The system architecture of the proposed system is shown in Figure 4. The proposed system consists of the following entries: MIT-DARPA 1999 data set preparation, network traffic data to ARFF conversion, association rules procedure, and Snort IDS rule generator procedure.

4.1 MIT-DARPA 1999 Data Set Preparation

To evaluate the performance of the proposed system, we utilized the MIT-DARPA 1999 data set [26]. However, the data sets were recorded separated by one week, named as the 1st week, 2nd week, 3rd week, 4th week, and 5th week datasets, respectively.

We utilized the *inside.tcpdump* and *outside.tcpdump* data sets of the 4th and 5th week datasets, as they contained both normal and abnormal behavior patterns. Additionally, these datasets contain network probe attacks (portsweep, satan, msscan, queso, ipsweep, etc.). This paper focuses on the ipsweep type only. We installed the Snort IDS version 2.9.2.2 [19] and a Mysql database, based on the CentOS 6.4 operating system [27]. As in a previous work [3], we improved the Snort IDS rules to record the network traffic data entries into the network traffic database found in the next module.

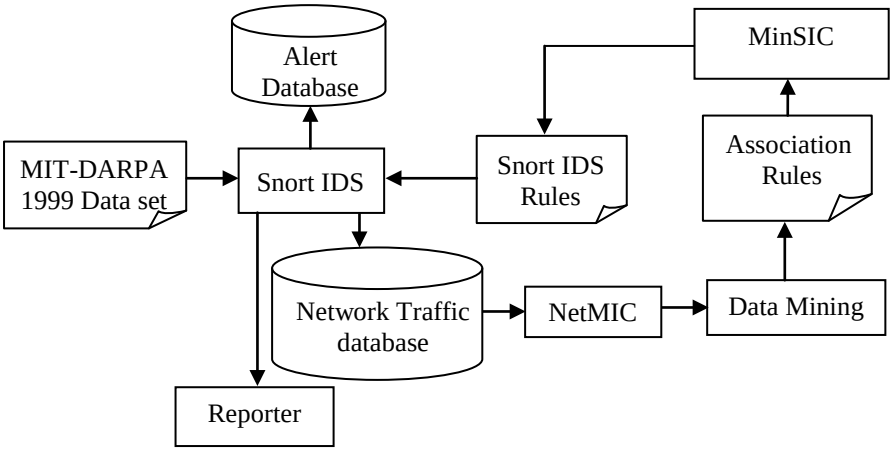

Figure 4 System Architecture.

4.2 Network traffic data to ARFF convertor

After the network traffic data have been recorded into the network traffic database, they are converted to *arff* format in the NetMIC module, before entering the data mining process with the Apriori algorithm [3].

Figure 5 Association rules with Aprori algorithm.

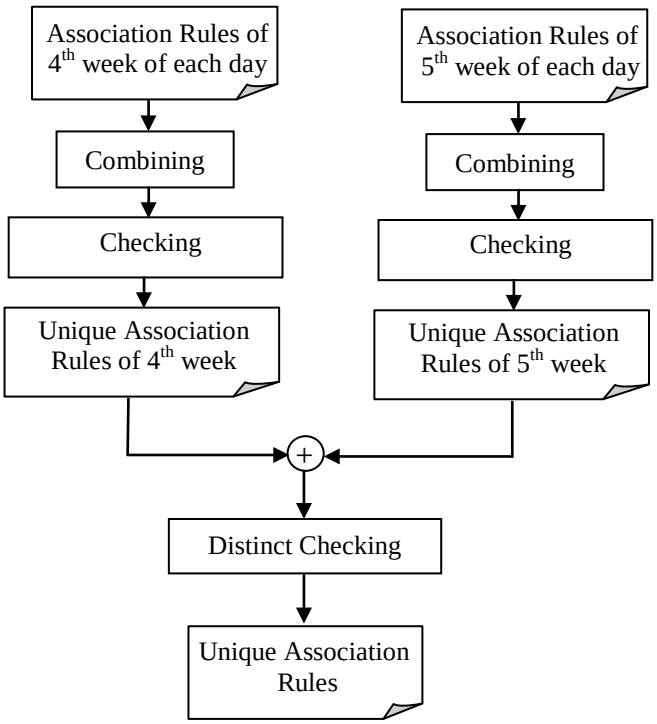

Figure 6 Association rules detect procedure.

4.3 Association Rules with Apriori Algorithm Procedure

The association rules are generated by defining the dissimilar parameters of the attribute number, the minimum support value (Min_sup), and the minimum confidence value (Min_conf), as shown in Figure 5. However, after the association rules have been processed some rules will maintain the same attribute, yet are placed in a different position. The same attributes remain distinct from each other, as shown in Figure 6. The association rules of each day of the 4th week were then combined. If similar association rules were found, only one was selected (likewise with each day of the 5th week). After this the unique association rules of the 4th week and 5th week were combined. Again, if similar association rules were found, only one was selected. The association rules were then used to generate the Snort IDS rules.

Table 1 shows the number of rules from the association rules technique of the ICMP protocol within several parameter conditions. There exists a negative correlation between the minimum support values (Min_sup %) and the total number of association rules (total rules). That is to say, when the minimum support value increases, the total number of association rules decreases. And

likewise, as the number of attributes increases, the maximum Min_sup (%) decreases as well as the total number of association rules.

Table 1 Association rules number of the ICMP protocol.

Attribut	Min_sup (%)	Min_conf (%)	Total Rules
6	10	100	448
	20	100	194
	30	100	106
	40	100	49
	50	100	49
	60	100	8
	70	100	1
	80	100	1
7	10	100	211
	20	100	88
	30	100	52
	40	100	15
	50	100	15
	60	100	1
8	10	100	61
	20	100	26
	30	100	15
	40	100	2
	50	100	2
9	10	100	8
	20	100	3
	30	100	2

4.4 Snort IDS Rules Generator Procedure

Figure 7 shows the Snort IDS rules generator procedure. The association rules of the defined parameter's entries are used to generate the Snort IDS rules by the MinSIC module for detecting network probe attacks. Figure 8 shows an example of a Snort IDS rules generator. In this example, the source address, destination address, source port, and destination port are assigned as "any".

5 Performance Evaluations

In this section, we describe the procedure of the performance evaluation of the proposed system. The performance evaluation is divided into two parts: classified network probe attacks (ipsweep type) and Snort IDS rules evaluation.

5.1 Classified Network Probe Attacks The Ipsweep Type

In this research, both the *inside.tcpdump* and *outside.tcpdump* files of the 4th week and 5th week dataset from the MIT-DARPA 1999 datasets were utilized for the performance evaluation. Additionally, they consist of DOS, U2R, R2L,

and network probe attacks. In previous studies, classifications of probe attacks included *portsweep*, *ipsweep*, *Satan*, *Is_domain*, *ntinfoscan*, and *queso* [5]. This paper, however, focuses only on the *ipsweep* type of network probe attacks.

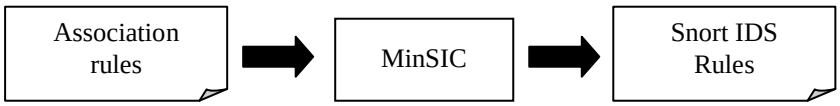

Figure 7 Snort IDS rules generator procedure.

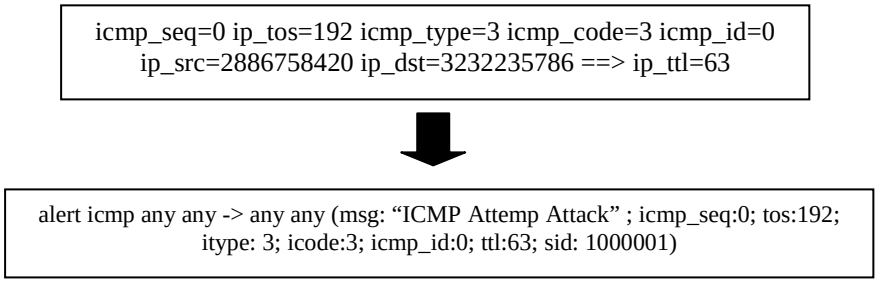

Figure 8 Transformed Snort IDS rules with MinSIC module.

Table 2 Summary of the network probe attack of the *ipsweep* type.

Week	Mon		Tue		Wed		Thu		Fri		Total Attacks
	In	Out	In	Out	In	Out	In	Out	In	Out	
4	0	-	-	0	0	0	0	10	0	268	278
5	0	5	0	0	0	0	0	10	0	-	15

Table 2 summarizes the *ipsweep* type of network probe attacks. We can see that the network probe attacks on Thursday and Friday of the 4th week in the *outsde.tcpdump* (out) file produced a total of 278 attacks. Moreover, Monday and Thursday of the 5th week in the *outside.tcpdump* file produced a total of 15 attacks. In some circumstances, Monday of the 4th and Friday of the 5th week in the *outside.tcpdump* file, as well as Tuesday of the 4th week in the *inside.tcpdump* file, we were unable to extract files for evaluation of the proposed system.

5.2 Snort IDS Rules Evaluation

The Snort IDS rules performances were evaluated by utilizing the accuracy comparisons procedure, previously described in [5]. Suitable parameters, as seen in Table 1, outline the Snort IDS rules as shown in Figure 7. We compared and evaluated the performance of the experimental results of the intrusion

detection between the original Snort IDS rules, *icmp.rules*, and *icmp-info.rules* of the Snort IDS, and the proposed Snort IDS rules.

We extrapolated an equation from [28] in order to calculate the accuracy rate of our proposed system, which is expressed as a percentage. The True Positive Percentage (TPP) is expressed in Eq. (3). A true positive is the detection of abnormal traffic that is accurately and correctly detected. A higher TPP value indicates better performance.

$$\text{True Positive Percentage} = \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100 \tag{3}$$

The True Negative Percentage (TNP) is expressed in Eq. (4). A true negative is the detection of normal traffic that is detected as a normal traffic. A higher TNP indicates better performance.

$$\text{True Negative Percentage} = \frac{\text{TN}}{\text{TN} + \text{FP}} \times 100 \tag{4}$$

The False Positive Percentage (FPP) is expressed in Eq. (5). A false positive is normal traffic that is incorrectly detected as abnormal traffic. A lower FPP indicates better performance.

$$\text{False Positive Percentage} = \frac{\text{FP}}{\text{FP} + \text{TN}} \times 100 \tag{5}$$

The False Negative Percentage (FNP) is expressed in Eq. (6). A false negative is abnormal traffic that is incorrectly detected as normal traffic. A lower FNP indicates better performance.

$$\text{False Negative Percentage} = \frac{\text{FN}}{\text{FN} + \text{TP}} \times 100 \tag{6}$$

Accuracy is expressed in Eq. (7), which is used to calculate the proposed system performance.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \times 100 \tag{7}$$

Table 3 summarizes the performance evolution of each parameter condition of the Snort IDS rules from the association rules data mining technique for the 4th week. The performance of each parameter condition is explained as follows:

Regarding TPP when the parameters were assigned 6, 7, and 8 attributes, Min_sup was 10% and Min_conf was 100%. The Snort IDS successfully detected the network probe attacks, as the TPP achieved was 100%. On the

other hand, when the parameter was assigned 7 and 8 attributes, the Snort IDS detection of network probe attacks achieved a TPP of 98.20%.

The TNP with a parameter of 8 attributes (Min_sup at 10% and Min_conf at 100%) achieved the highest performance, with 98.67%. However, at 7 attributes, the TNP performed the lowest, with 43.96%.

The FPP with a parameter of 8 attributes (Min_sup at 10% and Min_conf at 100%) achieved the highest performance, with 1.32%. When the parameter was assigned 7 attributes, the FPP scored the lowest performance, with 56.04%.

Table 3 Summary of the network traffic data detection on 4th week.

Attribute	Min_sup (%)	Min_conf (%)	TPP (%)	TNP (%)	FPP (%)	FNP (%)	Accuracy (%)
6	10	100	100.00	49.23	50.77	0.00	50.44
	20	100	0.00	47.32	52.68	100.00	46.20
	30	100	0.00	52.96	47.04	100.00	51.70
	40	100	0.00	54.99	45.01	100.00	53.68
	50	100	0.00	54.99	45.01	100.00	53.68
7	10	100	98.20	43.96	56.04	1.80	45.24
	20	100	0.00	54.17	45.83	100.00	52.89
	30	100	0.00	54.99	45.01	100.00	53.68
8	10	100	98.20	98.68	1.32	1.80	98.67

Table 4 Summary of the network traffic data detection on 5th week.

Attribute	Min_sup (%)	Min_conf (%)	TPP (%)	TNP (%)	FPP (%)	FNP (%)	Accuracy (%)
6	10	100	100.00	61.38	38.62	0.00	61.38
	20	100	0.00	63.35	36.65	100.00	63.31
	30	100	0.00	63.32	36.68	100.00	63.28
	40	100	0.00	72.12	27.88	100.00	72.07
	50	100	0.00	72.12	27.88	100.00	72.07
7	10	100	66.67	62.54	37.46	33.33	62.53
	20	100	0.00	62.47	32.53	100.00	67.43
	30	100	0.00	72.12	27.88	100.00	72.07
8	10	100	66.67	98.72	1.28	33.33	98.70

At a parameter of 6 attributes (Min_sup at 10% and Min_conf at 100%), the FNP achieved the highest performance with 0.00%. At both 7 and 8 attributes the FNP archived a slightly lesser performance of 1.80%. At all other Min_sup values, the FNP received the lowest performance, with 100%.

In calculating the accuracy of the proposed system's performance (Min_sup at 10% and Min_conf at 100%), the Snort IDS achieved the highest performance, with 98.67%, when assigned 8 attributes. Conversely, with a parameter of 7 attributes, the Snort IDS achieved the lowest performance, with 45.24%.

Table 4 summarizes the performance evolution of the network data detection for the 5th week. The performance of each parameter condition is explained as follows:

The TPP with 6 attributes assigned as the parameter (Min_sup at 10% and Min_conf at 100%) detected all network probe attacks, as the TPP archived 100%. The results, similar to that of week 4, differed when the parameters of 7 and 8 attributes were assigned. TPP performance achieved 66.67%.

Regarding TNP when the parameter was assigned 8 attributes (Min_sup at 10% and Min_conf at 100%), the highest performance achieved was 98.72%. A parameter of 7 attributes achieved the lowest performance, with 61.38% (versus 43.96% in week 4).

The FPP when assigned a parameter of 8 attributes (Min_sup at 10% and Min_conf at 100%) achieved the highest performance, with 1.28%. The lowest performance, 37.46%, was achieved with a parameter of 7 attributes.

FNP with 6 attributes (Min_sup at 10% and Min_conf at 100%) achieved the highest performance (similar to that of week 4) with 0.00%. However, with parameters assigned at 7 and 8 attributes, the FNP scored the lowest performance, with 33.33%.

In calculating the accuracy of the proposed system's performance in the fifth week with 8 attributes (Min_sup at 10% and Min_conf at 100%), the Snort IDS achieved the highest performance (almost identical to that of the 4th week) with 98.67%. Similar to the results of the 4th week, with a parameter of 7 attributes, the Snort IDS again achieved the lowest performance, this time with 45.24%.

Table 5 illustrates the performance evaluation of the original Snort IDS rules, comparing the 4th and 5th week datasets. The original Snort IDS rules' accuracy was very high (and very similar) in both the 4th and the 5th week, at 98.36% and 99.77% respectively. Similarities also occurred in the TNP of the 4th week (99.78%) and the 5th week (99.81%). However, the original Snort IDS rules were unable to detect all network probe attacks. The TPP was 0.00% in both the 4th and the 5th week, and the FPP of the 4th and the 5th week were 0.22% and 0.19% respectively. The FNPs of the 4th and the 5th week were both 100%.

Table 5 Performance evaluation of original Snort IDS rules.

Week	TPP (%)	TNP (%)	FPP (%)	FNP (%)	Accuracy (%)
4	0.00	99.78	0.22	100.00	98.36
5	0.00	99.81	0.19	100.00	99.77

Table 6 Performance evaluation of *icmp.rules* and *icmp-info.rules* of Snort IDS.

Week	TPP (%)	TNP (%)	FPP (%)	FNP (%)	Accuracy (%)
4	100.00	47.36	52.64	0.00	47.99
5	100.00	46.34	53.66	0.00	46.35

Table 7 Performance comparison of proposed and original system.

	Week	Original Snort IDS rules (%)	icmp.rules and icmp-info.rules of the Snort IDS rules (%)	Proposed Snort-IDS rules (%)
TPP	4	0.00	100.00	98.20
	5	0.00	100.00	66.67
TNP	4	99.78	47.36	98.68
	5	99.81	46.34	98.72
FPP	4	0.22	52.64	1.32
	5	0.19	53.66	1.28
FNP	4	100.00	0.00	1.80
	5	100.00	0.00	33.33
Accuracy	4	98.36	47.99	98.67
	5	99.77	46.35	98.70

Table 6 evaluates the performance of the *icmp.rules* and the *icmp-info.rules* of the Snort IDS for the 4th and 5th week datasets. The TPP of both the 4th week and the 5th week achieved 100%. However, the FPP scored low performances for both weeks with 52.67% and 53.66%, respectively. This implies that normal traffic was erroneously detected as abnormal traffic.

Table 7 shows a performance comparison of the original system, incorporating the original Snort IDS rules as well as the *icmp.rules*, and *icmp-info.rules* (within the Snort IDS rules) with the proposed system. The proposed system applied the association rules technique of data mining. In the comparison of all Snort IDS rules, we found that by assigning a parameter of 8 attributes (Min_sup at 10% and Min_conf at 100%), the proposed system achieved the greatest accuracy.

The performance evaluation found that the proposed system achieved a proximate accuracy equal with that of the original Snort IDS rules, as well as higher performance in both TPP and FNP. The proposed system also achieved higher performances in TNP and FPP, as well as greater accuracy compared with the *icmp.rules* and *icmp-info.rules* of the Snort IDS.

6 Conclusion and Future Work

Intrusion detection systems are efficient network security tools for detecting and monitoring network traffic data. They generate an alert when abnormal behavior patterns are matched to existing rules. However, because the IDS may have high false positive and false negative values, we have proposed an alternative system, incorporating data mining of the association rules within the Snort IDS. The proposed system was thoroughly tested and compared to the original Snort IDS Rules as well as *icmp.rules* and *icmp-info.rules* within the Snort IDS, the proposed system proved to be more efficient and more accurate. In future work, we will explore other data mining techniques for intrusion detection systems for DoS, U2R, and R2L.

References

[1] Zhimin, Z., Chen, Z., Zhou, T. & Guan, X., *The Study On Network Intrusion Detection System of Snort*, in Proceedings of The 2nd International Conference on Network and Digital Society (ICNDS), Wenzhou, China, Hong Kong Section CAS/COM Joint Chapter, Guizhou University, Peking University, **2**, pp. 194-196, 2010.

[2] Sonawane, S., Pardeshi S. & Prasad, D., *A Survey on Intrusion Detection Technique*, World Journal of Science and Technology, **2**(3), pp. 127-133, 2012.

[3] Khamphakdee, N., Benjamas, N. & Saiyod, S., *Network Traffic Data to ARFF Convertor for Association Rules Technique of Data Mining*, in Proceedings of The 5th IEEE Conference on Open System (ICOS), Subang Jaya, Malaysia, IEEE Malaysia Computer Chapter, pp. 89-93, 2014.

[4] *Intrusion Detection Attacks Database,* http://www.ll.mit.edu/mission/communications/cyber/CSTcorpora/ideval/docs/attackDB.html (30 June 2013).

[5] Khamphakdee, N., Benjamas, N. & Saiyod, S., *Improving Intrusion Detection System Based on Snort Rules for Network Probe Attack Detection*, in Proceedings of The 2nd International Conference on Information and Communication Technology (ICOICT), Bandung, Indonesia, Telkom University, IEEE Indonesia Section, pp. 69-74, 2014.

[6] Sandhu, U.A., Haider, S., Naseer, S. & Ateeb, O.U., *A Survey of Intrusion Detection & Prevention Technique*, International Conference on Information Communication and Management, Singapore, IACSIT Press, **16**, pp. 66-71, 2011.

[7] Naiping, S. & Genyuan, Z., *A Study on Intrusion Detection Based on Data Mining*, in Proceedings of International Conference of Information Science and Management Engineering (ISME), Xi'an, China, IEEE Computer Society, **1**, pp. 135-138, 2010.

[8] Pu, W. & Jun-Qing, W., *Intrusion Detection System with the Data Mining Technologies*, in Proceedings of The 3rd IEEE International Conference on Communication Software and Network (ICCSN), Xi'an, China, Xidian University, IEEE Beijing Section, IEEE Xian Section, pp. 490-492, 2011.

[9] Xue, M. & Zhu, C., *Applied Research on Data Mining Algorithm in Network Intrusion Detection*, in Proceedings of International Joint Conference on Artificial Intelligence (JCAI '09), Hainan, Island, Intelligent Information Technology Application Association (iita), IEEE Computer Society, pp. 275-277, 2009.

[10] Denatious, D.K. & John, A., *Survey on Data mining Techniques to Enhance Intrusion Detection*, in Proceedings of International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, Sri Shakthi Instirute of Engineering & Technology Student Branch, IEEE Madras Section, pp. 1-5, 2012.

[11] Kumar, V. & Sangwan, O.P., Signature Based Intrusion Detection System Using SNORT, International Journal of Computer Applications & Information Technology, **1**(3), pp. 35-41, 2012.

[12] Shah, S.N. & Singh P., *Signature-Based Network Intrusion Detection System Using SNORT And WINPCAP*, International Journal of Engineering Research & Technology (IJERT), **1**(10), pp. 1-7, 2012.

[13] Geng, X., Liu, B. & Huang, X., *Investigation on Security System for Snort-Based Campus Network*, in Proceedings of The 1st International Conference on Information Science and Engineering (ICISE), Nanjing, China, Nanjing University of Science and Technology, IEEE Nanjing Section, pp. 1756-1758, 2009.

[14] Rani, S. & Singh V., *SNORT: An Open Source Network Security Tool for Intrusion Detection in Campus Network Environment*, International Journal of Computer Technology and Electronics Engineering, **2**(1), pp. 137-142, 2012.

[15] Huang, C., Xiong, J. & Peng, Z., *Applied Research on Snort Intrusion Detection Model in The Campus Network*, in Proceedings of IEEE Symposium on Robotics and Applications (ISRA), Kuala Lumpur, Malaysia, IEEE Malaysia Section IE/IA/PEL Joint Chapter, pp. 596-599, 2012.

[16] Haixia, G., *Research of the Intrusion Detection System Based On Data Mining*, in Proceedings of International Conference on e-Education, Entertainment and e-Management (ICEEE), Bali, Indonesia, International Association of Management Science and Industrial Engineering, IEEE Indonesia Section, pp. 190-192, 2011.

[17] Miao, C. & Chen, W., *A Study of Intrusion Detection System Based on Data Mining*, in Proceedings of IEEE International Conference on Information Theory and Information Security (ICITIS), Beijing, China, Beijing University of Posts and Telecommunications, IEEE Beijing Section, pp. 186-189, 2010.

[18] Wu, G. & Huang, Y., *Design of A New Intrusion Detection System Based on Database*, in Proceedings of International Conference on Signal Processing System, Singapore, International Association of Computer Science and Information Technology, IEEE Computer Society, pp. 814-817, 2009.

[19] Snort, https://snort.org (5 April 2013).

[20] Roesch, M., *Snort-Lightweight Intrusion Detection for Networks*, in Proceedings of LISA'99: 13th Systems Administration Conference, Washington, USA, pp.229-238, 1999.

[21] Xu, J., Zhaug, J., Gadipalli, T., Yuan, X. &Yu, H., *Learning Snort Rules by Capturing Intrusions in Live Network Traffic Replay*, in Proceedings of The 15th Colloquium for Information System Security Education, Fairborn, Ohio, USA, The Colloquium for Information Systems Security Education, pp. 145-150, 2011.

[22] Rehman, R.U., *Intrusion Detection System with Snort*, Prentice Hall PTR Upper Saddle River, New Jersey, 2003.

[23] Zhao, Q. & Bhowmick, S.S., *Association Rule Mining: A Survey*, Nanyang Technological University, Singapore, https://www.lri.fr/~antoine/Courses/Master-ISI/Regle-association.pdf (15 June 2014).

[24] Agrawal, R., Imielinski, T. & Swami, A., *Mining Association Rules Between Sets of Items in Large Databases*, in Proceedings of The ACM SIGMOD Conference on Management of Data, Washington, D.C., USA, Association for Computing Machinery (ACM), pp. 207-216,1993.

[25] Agrawal, R. & Srikant, R., *Fast Algorithm for Mining Association Rules*, in Proceedings of The 20th International Conference on Very Large Data Bases (VLDB), Santiago, Chile, Association for Computing Machinery (ACM), pp. 487-499, 1994.

[26] MIT-DARPA 1999 Data Set, http://www.ll.mit.edu/mission/communications/cyber/CSTcorpora/ideval/data/1999data.htm (30 June 2013).

[27] Linux CentOS, http://www.centos.org (1 April 2013).

[28] Elshoush, H.T. & Osman, I.M., *Alert Correlation in Collaborative Intelligent Intrusion Detection Systems-A Survey*, Applied Soft Computing, **11**(7), pp. 4349-4365, 2011.